AT&T at TREC-7 SDR Track

Amit Singhal, John Choi, Donald Hindle, Julia Hirschberg, Fernando Pereira, Steve Whittaker

AT&T Labs–Research
180 Park Avenue
Florham Park, NJ 07932

ABSTRACT

AT&T participated in the Spoken Document Retrieval (SDR) track of TREC-7. Our speech retrieval system uses modern Information Retrieval (IR) methods in conjunction with in-house automatic speech recognition. The novel feature of our TREC-7 work is the use of *document expansion* to reduce the performance loss due to ASR errors. Results show that retrieval from automatic transcriptions of speech is quite competitive with doing retrieval from human transcriptions. Our experiments indicate that document expansion can be used to further improve retrieval from automatic transcripts. This paper presents some analysis of document expansion in context of the TREC-7 SDR track task.

1. INTRODUCTION

Text REtrieval Conference, or TREC for short, is an annual series of evaluation workshops organized by NIST and DARPA to evaluate modern text retrieval and related technologies on large scale datasets. The seventh Text REtrieval Conference (TREC-7) was held at NIST on November 9–11, 1998. [10] TREC-7 included a track on spoken document retrieval (SDR track) that was used to evaluate how modern IR technology can be used in conjunction with modern speech recognition technology to search spoken content. Eight teams participated in the *full SDR* (recognition as well as retrieval) category of the track[1]. AT&T participated in the full SDR category. [9]

We use an internal speech recognition system based on weighted finite-state transducers. [4]. Our IR system is an internally modified version of Cornell's well-known SMART retrieval system. [1, 6] For speech retrieval, we believe that parallel text corpora, for example printed news from the same time period, can be successfully exploited to improve retrieval effectiveness of a system. This is especially true for the news material currently being used in the SDR track. We use these ideas in our SDR track participation. Initial results from the use of a parallel corpus are quite encouraging.

2. SPEECH RECOGNIZER

Our speech recognition process involves the following steps. Prior to recognition, each speech story is segmented into approximately one minute long prosodically well-formed segments using a CART based classifier. [2] The resulting segments are submitted to another wideband/narrowband classifier for selection of the acoustic model to be used in recognition of that segment.

The recognizer is based on a standard time-synchronous beam search algorithm. The probabilities defining the transduction from text-dependent phone sequences to word sequences are estimated on word level grapheme-to-phone mappings and are implemented in the general framework of weighted finite-state transducers. [4] Transducer composition is used to generate word lattice output.

We use continuous density, three-state, left-to-right, context-dependent hidden Markov phone models. These models were trained on 39-dimensional feature vectors consisting of the first 13 mel-frequency cepstral coefficients and their first and second time derivatives. Training iterations included eigenvector rotations, k-means clustering, maximum likelihood normalization of means and variances and Viterbi alignment. The output probability distributions consist of a weighted mixture of Gaussians with diagonal covariance, with each mixture containing at most 12 components. The training data were divided into wideband and narrowband partitions, resulting in two acoustic models.

2.1. Language Models

We used a two pass recognition process. In the first pass, we built word lattices for all the speech using a minimal trigram language model and a beam that we had determined heuristically to provide manageable word lattices. These word lattices were then rescored, by removing the trigram grammar weights while retaining the acoustic weights and intersecting these lattices with a 4-gram language model. The 1-best path was extracted from the rescored lattices.

Both the first pass trigram language model and the rescoring 4-gram model are standard Katz backoff models [3], using the same 237 thousand word vocabulary. For choosing the vocabulary, all of the words from the SDR98 training transcript were used. This base vocabulary was supplemented with all words of frequency greater than two appearing in the New York Times and LA Times segments of LDC's North American News corpus (LDC Catalog Number: LDC95T21, see `www.ldc.upenn.edu`), in the period from June 1997 through January 1998. The vocabulary includes about 5,000 common acronyms (e.g "N.P.R."), and the training texts were preprocessed to include these acronyms.

The language model training was based on three transcription sources (the SDR98 training transcripts, HUB4 transcripts, transcripts of NBC nightly news) and one print source (the LDC NA News corpus of newspaper text). The first-pass trigram model was built by first constructing a backoff language model from the 271 million words of training text, yielding 15.8 million 2-grams and 22.4 million 3-grams. This model was reduced in size, using the approach of Seymore and Rosenfeld [7], to 1.4 million 2-grams and 1.1 million 3-grams. When composed with the lexicon, this smaller trigram model yielded a manageable sized network. The second pass

[1] The quasi SDR category allowed teams to use the recognition output of CMU's SPHINX-III system instead of their our recognition, and three teams participated under this category.

d *tf* factor:

$$1 + ln(1 + ln(tf)) \qquad 0 \ if \ tf = 0$$

t *idf* factor:

$$log(\frac{N+1}{df})$$

b pivoted byte length normalization factor:

$$\frac{1}{0.8 + 0.2 \times \frac{length\ of\ document\ (in\ bytes)}{average\ document\ length\ (in\ bytes)}}$$

where, *tf* is the term's frequency in text
N is the total number of documents
df is # of documents containing the term

dnb weighting: **d** factor × **b** factor
dtb weighting: **d** factor × **t** factor × **b** factor
dtn weighting: **d** factor × **t** factor

Table 1: Term Weighting Schemes

model used 6.2 million 2-grams, 7.8 million 3-grams, and 4.0 million 4-grams. For this model, the three transcription sources (SDR, HUB4, NBC) were in effect interpolated with the text source (NA News), with the latter being give a weight of 0.1. *The word error rate for our recognizer for the SDR track data was 31%.* These transcriptions are labeled *ATT-S1*.

3. RETRIEVAL SYSTEM

For the SDR track, we use the NA News corpus (also used in the language model training described above) as the parallel corpus for query and document expansion (described below). Since the test data is dated from June 1997 to January 1998, we used news dated from May 1997 to February 1998 (one month before and after) from the NA news corpus.

3.1. The Task

In the SDR track, participants had to search a collection of 100 hours of speech recordings for documents given 23 different user queries. An example user query is (query # 71) *"What government officials have been convicted of a crime?"*. The speech recordings were manually segmented into 2,866 different stories, and each story was judged by the user as being relevant or irrelevant to his/her query. The aim was then to, given a user query, rank these stories using an IR system so that most relevant stories are ranked before most non-relevant stories. This task has often been called the ad-hoc searching task in the TREC community. However, the difference from a standard text retrieval task is the use of erroneous automatic speech transcriptions for the stories in place of perfect text.

Like most other participants, we create word-level transcriptions for these stories using our recognizer and use our ad-hoc searching algorithm to do retrieval over these erroneous transcriptions. The effectiveness of a ranking is measured via non-interpolated average precision, a standard metric used in IR to measure retrieval effectiveness. More details on the ad-hoc task and its evaluation can be found in [10].

	Base	Query Expansion
Human	0.4595	0.5300 (+15.3%)
ATT-S1	0.4353	0.5020 (+15.3%)
	(–5.3%)	(–5.3%)

Table 2: Average precision results.

3.2. Retrieval Algorithm

Even though we used a slightly different algorithm in our official TREC-7 participation, using the following algorithm, which is what we use in the main TREC ad-hoc task, yields consistently better results. Here are the main steps in the algorithm.

- **Pass-1**: Using *dtn* queries and *dnb* weighted documents (see Table 1), a first-pass retrieval is done.
- *Expansion:* Top ten documents retrieved in the first pass are *assumed* to be relevant to the query and documents ranked 501–1000 are assumed to be non-relevant. Rocchio's method (with parameters $\alpha = 3,\ \beta = 2,\ \gamma = 2$) is used to expand the query by adding twenty new words and five new phrases with highest Rocchio weights. [5] To include the *idf*-factor in the expansion process, documents are *dtb* weighted.
- **Pass-2:** The expanded query is used with *dnb* documents to generate the final ranking of 1,000 documents.

Table 2 shows that retrieval from automatic transcriptions with 31% WER is about 5% worse than retrieval from perfect transcriptions. We also see that the query expansion step improves the retrieval effectiveness noticeably, by over 15%. These results are important as they show the viability of doing very effective speech retrieval using modern speech recognition and IR technologies.

4. DOCUMENT EXPANSION

The one-best transcript from a recognizer misses many content words and adds some spurious words to a spoken document. The misses reduce the *word-recall* (proportion of spoken words that are recognized) and the spurious words reduce the *word-precision* (proportion of recognized words that were spoken). We believe that information retrieval algorithms would benefit from a higher word recall and are robust against poor word precision. An approach to enhance word recall is to add new words that "could have been there" (words that were probably spoken but weren't the top choice of a speech recognizer) to the automatic transcriptions of a spoken document.

Several techniques are plausible for bringing new words into a document. An obvious one from an IR perspective is *document expansion* using similar documents: find some documents related to a given document, and add new words from the related documents to the document at hand. And from a speech recognition perspective, the obvious choice is to use word lattices which contain multiple recognition hypotheses for any utterance. A word lattice contains words that are acoustically similar to the recognized words could have been said instead of the words recognized in the one-best transcription.

In our official TREC-7 participation we used a constrained document expansion which used only those expansion words that are

Code	Provided By	WER
Human	NIST	0%
CUHTK-S1	Cambridge University	24.8%
Dragon98-S1	Dragon Systems	29.8%
ATT-S1	AT&T Labs	31.0%
NIST-B1	Carnegie Melon (CMU)	34.1%
SHEF-S1	Sheffield University	36.8%
NIST-B2	Carnegie Mellon (CMU)	46.9%
DERASRU-S2	DERA	61.5%
DERASRU-S1	DERA	66.2%

Table 3: Different automatic transcriptions.

Transcript	**Unexpanded Docs**		**Expanded Docs**	
	Base	Qry Expn	Base	Qry Expn
Human	0.4595	0.5300	0.5108	0.5549
CUHTK-S1	0.4376	0.5035	0.5220	0.5372
Dragon98-s1	0.4190	0.5100	0.5061	0.5284
ATT-S1	0.4353	0.5020	0.5080	0.5343
NIST-B1	0.4104	0.4820	0.4862	0.5259
SHEF-S1	0.4073	0.4890	0.5068	0.5421
NIST-B2	0.3352	0.3965	0.4377	0.4743
DERASRU-S2	0.3633	0.3962	0.4585	0.5065
DERASRU-S1	0.3236	0.3613	0.4526	0.4849

Table 4: Cross-recognizer analysis.

proposed by similar documents and also appear in a word-lattice. However, after the official conference we did a more rigorous study of document expansion and discovered that constraining document expansion to allow only terms from the word-lattices generated by our recognizer held no additional benefit over not doing so. *I.e.* we can do document expansion only from NA news and the results were equally good or better. This also allows us to test document expansion for retrieval from the automatic transcriptions provided by other SDR track participants, for which we don't have the word-lattices.

We test document expansion on different automatic transcriptions provided to NIST by various track participants. Table 3 lists these transcriptions along with their word error rates. Here are the steps involved in document expansion:

1. Find documents related to a speech document. We do this by running the automatic transcription of the speech document as a query (raw-*tf*$\times$*idf* weighted) on the NA News corpus and retrieving the *ten* most similar documents. In other words, we use the ten nearest neighbors of the speech document in this process. The documents are weighted by raw-*tf*$\times$*idf* when used as a query because we found that nearest neighbors found using raw-*tf*$\times$*idf* weighted documents yield the best expansion results.
2. The speech transcriptions are then modified using Rocchio's formula.
$$\vec{D}_{new} = \vec{D}_{old} + \frac{\sum_{i=1}^{10} \vec{D}_i}{10}$$
where $\vec{D}_{old}$ is the initial document vector, $\vec{D}_i$ the the vector for the i-th related document, and $\vec{D}_{new}$ is the modified document vector. All documents are *dnb* weighted (see Table 1). New words are added to the document. For term selection, the Rocchio weights for new words are multiplied by their *idf*, the terms are selected, and the *idf* is stripped from a selected term's final weight. Furthermore, to ensure that this document expansion process doesn't change the effective length of the document vectors, and change the results due to document length normalization effects, [8] we force the total weight for all terms in the new vector to be the same as the total weight of all terms in the initial document vector. We expand documents by 100% of their original length (*i.e.* if the original document has 60 indexed terms, then we add 60 new terms to the document).

The results for unexpanded as well as the expanded documents are listed in Table 4. The two main highlights of these results are:

- document expansion yields large improvements across the board, and more importantly
- document expansion reduces the performance gap between retrieval from perfect and automatic transcriptions.

These points are highlighted in Figure 1. The left plot shows the average precision on the y-axis, against the WER on the x-axis. All number plotted in Figure 1 are for the unexpanded queries (*i.e.* we use the columns marked *Base* in Table 4). This prevents effects of query expansion from affecting these graphs and allows us to study the effects of document expansion in isolation. The horizontal lines are for human transcriptions whereas the other lines are for the different automatic transcriptions. As we can see in the left graph, document expansion (solid lines) yields large improvements across the board for this task over not doing document expansion (dashed lines). This is indicated by the noticeably higher average precision for the solid lines as compared to the corresponding dashed lines.

The right graph in Figure 1 plots the %-loss from human transcriptions on the y-axis for unexpanded and expanded documents. The baseline for the expanded documents is higher; it is the expanded human transcriptions, *i.e.* the solid horizontal line on the left graph. We observe that for the poorest transcriptions (DERASRU-S1) document expansion yields an improvement of an impressive 40% (over 0.3236) and reduces the performance gap from human transcription to about 12% instead of the original 30% despite the higher baseline used. The results are similar for other transcriptions.

It might be the case that for this test collection document expansion is beneficial in general, and it doesn't hold any special advantage for automatic speech transcripts. However, the right graph in Figure 1 shows that this is not the case, and document expansion indeed is more useful when the text is erroneous. The dashed line on the right graph shows the loss in average precision when retrieval is done from (unexpanded) automatic transcriptions instead of (unexpanded) human transcriptions. This line has the same shape as the dashed line on the left graph since it is essentially the same curve on a different scale (0 to –100 in % loss, the human transcriptions being the 0% mark). And we notice that the loss for CUHTK-S1 (the leftmost point) is close to 0% whereas it is 30% for DERASRU-S1 (the rightmost point). The solid line on the right plot shows the losses for various transcripts for expanded documents. The baseline for this curve is higher; it corresponds to the solid horizontal line on the

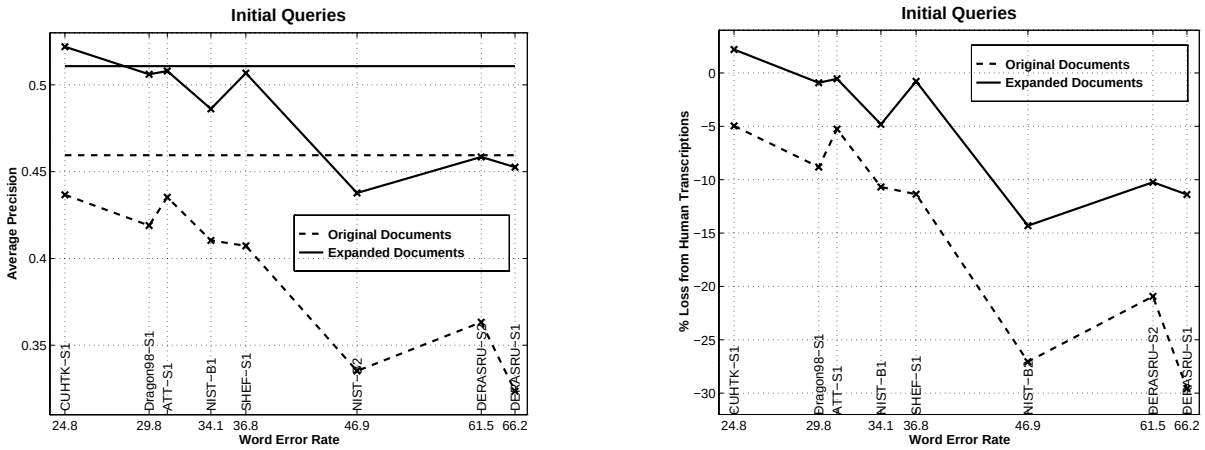

Figure 1: Raw average precision and %-loss from human transcriptions (initial user queries).

left graph. We see that document expansion indeed benefits the poor transcriptions much more that it benefits the human or the better automatic transcriptions. For poor transcriptions, the gap in retrieval effectiveness reduces from 27% to about 15% for NIST-B2, from 22% to about 10% for DERASRU-S2, and from about 30% to about 11% for DERASRU-S1. All these loss reductions are quite significant.

In summary, document expansion is more useful for automatic speech transcripts than it is for human transcriptions. Automatic recognitions that are relatively poor need the most help during retrieval. Document expansion is helping exactly these transcriptions, and quite noticeably. It is encouraging that even with word error rates as high as 65%, the retrieval effectiveness drops just 10-15% post document expansion. This drop would have been 22-30% without expansion.

5. CONCLUSIONS

These results establish the following two facts:

1. Given a reasonable speech recognition, retrieval effectiveness for automatic transcription is comparable to that for perfect transcriptions of speech. However, if the transcriptions are poor, we do get a very large drop in retrieval effectiveness.

2. Document expansion helps speech retrieval, but most importantly it helps retrieval noticeably when such help is badly needed, *i.e.* for very poor automatic speech transcriptions.

The second result is quite encouraging, and we will study this effect further.

6. ACKNOWLEDGMENTS

We are very grateful to all the SDR track participants who have provided their transcriptions to NIST. We are also thankful to NIST for making these transcriptions available to us. This work wouldn't have been possible without the support Andrej Ljolje and Michael Riley.

References

1. Chris Buckley. Implementation of the SMART information retrieval system. Technical Report TR85-686, Department of Computer Science, Cornell University, Ithaca, NY 14853, May 1985.

2. Julia Hirschberg and Christine Nakatani. Using machine learning to identify intonational segments. In *Proceedings of the AAAI Spring Symposium on Applying Machine Learning to Discourse Processing*, Palo Alto, CA, March 1998.

3. S.M. Katz. Estimation of probabilities from sparse data from the language model component of a speech recognizer. *IEEE Transactions of Acoustics, Speech and Signal Processing*, pages 400–401, 1987.

4. Fernando C. N. Pereira and Michael D. Riley. Speech recognition by composition of weighted finite automata. In Emmanuel Roche and Yves Schabes, editors, *Finite-State Language Processing*, pages 431–453. MIT Press, Cambridge, Massachusetts, 1997.

5. J.J. Rocchio. Relevance feedback in information retrieval. In *The SMART Retrieval System—Experiments in Automatic Document Processing*, pages 313–323, Englewood Cliffs, NJ, 1971. Prentice Hall, Inc.

6. Gerard Salton, editor. *The SMART Retrieval System—Experiments in Automatic Document Retrieval*. Prentice Hall Inc., Englewood Cliffs, NJ, 1971.

7. Kristie Seymore and Ronald Rosenfeld. Scalable backoff language models. In *ICSLP'96*, volume 1, 1996.

8. Amit Singhal, Chris Buckley, and Mandar Mitra. Pivoted document length normalization. In *Proceedings of the Nineteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 21–29. Association for Computing Machinery, New York, August 1996.

9. Amit Singhal, John Choi, Donald Hindle, David Lewis, and Fernando Pereira. AT&T at TREC-7. In E.M. Voorhees and D.K. Harman, editors, *Proceedings of the Seventh Text REtrieval Conference (TREC-7)*, 1999 (to appear).

10. E.M. Voorhees and D.K. Harman. Overview of the seventh Text REtrieval Conference (TREC-7). In E.M. Voorhees and D.K. Harman, editors, *Proceedings of the Seventh Text REtrieval Conference (TREC-7)*, 1999 (to appear).